웹 검색을 활용한 기사 표절 탐지 시스템

조정현○ 김유섭
한림대학교 컴퓨터공학과
{showcjh, yskim01}@hallym.ac.kr

A Plagiarism Detection System for Newspaper Articles by using Web Search

Junghyun Cho○ Yu-Seop Kim
Dept. of Computer Engineering, Hallym University

요 약

최근 문서 저작권에 대한 관심과 중요도가 높아지고 있고 문서 표절에 관한 연구도 지속적으로 이루어지고 있다. 최근 기사의 표절 또는 무단도용 문제가 적지 않게 발생하고 있다. 현재까지의 문서 표절 연구는 실시간 특성이 매우 강한 신문 기사의 표절 문제에 적용하기 어려웠다. 따라서 현재는 이러한 표절 기사를 가려내기 위해 수 많은 신문사에서 하루 수천 건씩 올라오는 기사들을 눈으로 일일이 가려내는 상황이다. 본 논문에서는 이러한 시간과 비용의 문제를 줄이기 위해 네이버와 다음에서 제공하는 웹 검색 OpenAPI를 활용해 표절 가능성이 있는 기사들을 자동으로 탐지해 내는 시스템을 제안한다. 제안하는 시스템은 하나의 원본 기사에서 5개의 문장을 랜덤으로 추출하고 각각의 문장을 검색어(query)로 사용해 연동된 OpenAPI를 사용하여 웹에서 기사를 검색한다. 또한 5번의 검색에서 추출되는 URL의 검색 빈도를 계산하여 해당 기사의 표절 가능성을 사용자가 쉽게 예측 할 수 있도록 하였다.

1. 서 론

최근 문서 저작권에 대한 기준이 엄격해 지면서 표절 문제가 여러 분야에서 발생하고 있고 그 심각성도 날로 커지고 있다. 특히 인터넷 정보의 양이 기하급수적으로 늘어나고 개인이 운영하는 홈페이지, 블로그 등의 수가 급격히 증가함에 따라 인터넷상에서의 표절도 많아지고 있다. 특히, 블로그나 신문 기사 등에서 문서의 표절 또는 무단도용이 자주 발생되고 있다. 이에 본 논문에서는 신문 기사 표절을 탐지할 수 있는 방법을 제시하고자 한다. 그러나, 본 연구의 방법론은 블로그 표절 탐지에 있어서도 동일하게 적용될 수 있다.

현재까지, 문서 표절에 관한 연구는 지속적으로 진행되고 있다. 주로 프로그램 소스 코드 표절 연구[1], 영문 문서 표절 연구[2], 한글 문서 표절 연구[3-5]와 관련한 연구들이 진행되고 있다. 표절과 관련한 연구 중에서 [3-5]과 같이 그 대상이 한국어 텍스트 문서인 연구들은 먼저 영어와는 다른 한국어의 특성을 활용하여 표절 여부를 판단하는데 있어 매우 정교한 alignment 알고리즘을 사용하여 부분 표절 등에도 적용할 수 있는 장점을 가지고 있다. 그러나, 이러한 방법론들은 매일 수백개의 신문사에서 각각 수백건씩 새로이 게재되는 신문 기사의 표절 문제에는 그대로 적용하기에 적절치 않다. 마찬가지로 매일 수만건씩 올라오는 블로그 문서에도 또한 적용하기에 적절치 않다. 현재 이러한 표절 기사를 가려내기 위해서는 수 많은 신문사에서 하루 수천 건씩 올라오는 기사들을 눈으로 일일이 가려내고 있는 상황이다. 이러한 작업은 시간, 인력, 비용이 크게 소모될 뿐만 아니라 그 정확도 역시 담보할 수 없다.

이에 본 논문에서는 현재 인터넷으로 거의 모든 신문 기사를 볼 수 있고 또한 매일 실시간으로 기사가 업데이트 되고 있는 점을 고려하여 효율적인 표절 기사의 탐지를 위해 네이버(Naver)[1]와 다음(Daum)[2]에서 제공하는 OpenAPI(Open Application Program Interface)를 활용한 표절 기사 탐지 시스템을 제안한다. 또한 표절 기사 탐지 정확성과 효율성을 입증하기 위해 실제 여러 원본의 기사들을 정해놓고 검색되는 표절 기사를 확인하는 실험을 통해 성능을 측정한다.

본 연구에서는 연합뉴스[3]에서 새로이 게재된 기사를 이후 게재된 타 신문사의 기사들이 단순히 참조를 했는지 아니면 표절을 했는지를 판단하는 시스템을 제안하였다. 이를 위하여 연합뉴스의 기사 중에서 랜덤하게 5개의 구문을 추출하여 OpenAPI의 검색 질의로 선정하였다. 그리고 5번의 검색을 실행하여 검색된 기사들의 URL을 수집하였으며 이 URL 중에서

[1] http://openapi.naver.com/
[2] http://dna.daum.net/apis/
[3] http://www.yonhapnews.co.kr/

2번 이상 검색된 기사들을 표절 가능성이 높은 기사로 선정하였다.

2장에서는 신문 기사들의 표절 유형에 대하여 분석하였고, 3장에서는 본 논문에서 제안한 표절 탐지 시스템의 전체적인 구조에 대하여 설명하였다. 각 구성원들은 보다 상세히 설명되었다. 그리고 4장에서는 구현 및 성능 평가 결과에 대하여 설명하고, 마지막 5장에서 결론 및 향후 연구 방향에 대하여 논할 것이다.

2. 기사의 표절 유형

본 논문에서는 기사의 표절 유형을 다음 4가지로 분류하였다. 첫째, 원본기사의 내용을 그대로 가져와 표절하는 유형이다. 이 유형은 출처를 알리는 크래딧 없이 원래 기사와 똑 같은 내용을 무단전재 하는 것으로 작성된 기사가 마치 표절한 기자에 의하여 전적으로 게재된 것처럼 오인하도록 한다.

두 번째는 원본기사 중에 몇 개의 문장을 가져와 조합한 유형이다. 예를 들면 원본기사의 내용을 별도의 수정 없이 몇 개의 문장 만으로 축소하는 것이다. 보통은 마지막 문장 몇 개 또는 중간의 문장을 몇 개를 뺀다.

세 번째는 원본기사 중에 몇 개의 문장을 조금씩 고쳐서 사용하거나 기존 문장에 자신이 새로운 어휘를 조금씩 추가하는 유형이다. 그림 1은 원본 기사를 그리고 그림 2는 이 유형의 예시 문장을 보여준다. 그림 2의 문장들을 보면 원본 기사를 약간씩 수정하여 작성했음을 알 수 있다.

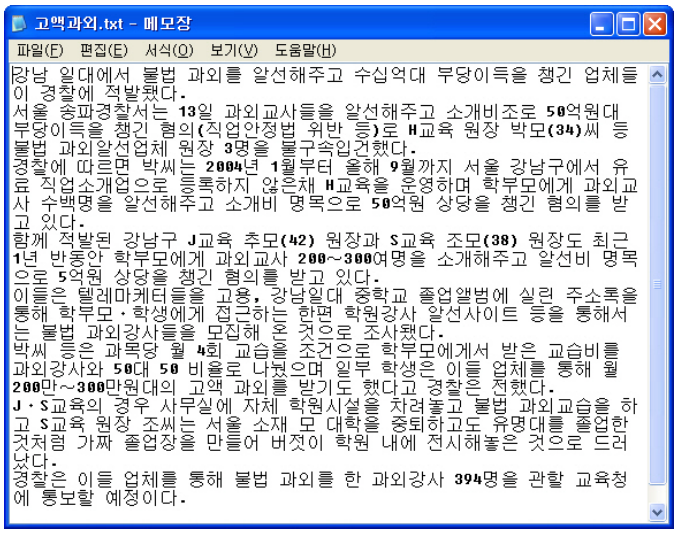

그림 1 기사 원본

마지막 네 번째는 원본기사의 부분 부분을 개조하고 자신이 쓴 내용과 섞어서 쓰는 유형이다. 그러나 이러한 유형은 실제 표절 여부를 판단하는데 있어 매우 전문적인 지식과 경험이 필요하기 때문에 본 연구에서는 이 유형의 표절 여부는 판단하지 않는다.

강남의 불법 고액과외 강사가 적발된 사례는 많지만 이번처럼 고액과외 강사를 학부모에 연결해 주는 브로커가 적발되는 것은 매우 드문 일이다. 이로써 그동안 소문으로만 떠돌던 고액과외 브로커의 존재가 실제하는 것으로 확인된 셈이다.

이에 지금까지 경찰 조사에서 드러난 박씨의 브로커 활동을 통해 강남고액과외 실태를 들여다봤다.

경찰에 따르면 박씨와 함께 적발된 강남구 J교육 추모(42) 원장과 S교육 조모(38) 원장도 최근 1년 반동안 학부모에게 과외교사 200~300여명을 소개해주고 알선비 명목으로 5억원 상당을 챙긴 혐의를 받고 있다.

그림 2 세 번째 유형의 표절 사례

3. 기사 표절 탐지 시스템

본 논문에서는 수많은 신문사에서 하루 수천 건씩 실시간으로 올라오는 신문 기사의 특성을 활용하기 위하여 기존의 웹 검색 업체에서 제공하는 OpenAPI를 기반하여 개발되었다. 본 연구에서 제안된 시스템의 전체 구조는 그림 3에서 볼 수 있다.

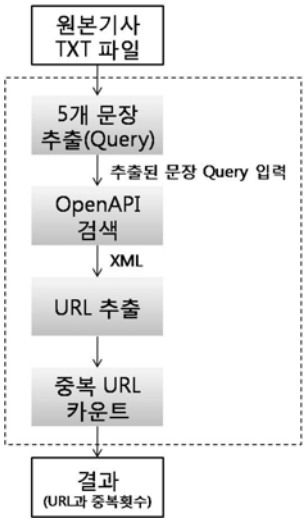

그림 3 시스템 전체 구성

본 시스템은 입력된 하나의 원본 기사에서 5개의 구문을 추출하고 추출된 각각의 구문을 질의어(query)로 사용해 OpenAPI로 각각 검색하여 검색된 문서들의 URL을 추출한다. 그리고 검색된 URL의 중복 횟수를 계산하여 해당 URL과 중복 횟수를 보여줌으로써 해당 문서의 표절 가능성을 가늠할 수 있도록 한다.

3.1 원본 기사

그림 1은 원본 기사의 유형을 보여준다. 원본 기사는 연합뉴스에서 매일 제공되는 실시간 기사로써, 일반적으로 연합뉴스는 이 기사를 일반 신문사에 실시간으로 제공한다. 인력 및 자원에 제한이 있는 많은 신문사들은 연합뉴스의 크래딧을 달고 이 기사들을 자사의 신문에 게재한다. 연합뉴스는 크래딧을 통하여 기사의 저작권이 자사에 있음을 명시한다는 조건에서

해당 기사의 전재 또는 부분 전재를 허용한다. 따라서 본 연구에서는 크래딧을 통하여 원저작권자를 명시하지 않고 무단으로 전체 및 부분 전재를 한 신문 기사를 탐지하는 것을 그 목적으로 한다.

3.2 질의 문장 추출

본 연구에서는 질의 문장을 원본 기사에서 랜덤하게 추출하는 것을 원칙으로 하였다. 또한 추출된 문장의 처음에서 6번째 어절까지를 따로 추출하여 질의어로 하였다. 본 연구에서 질의어의 길이를 6개로 정한 이유는 최근 논문 표절의 기준에 6개의 연속된 어절이 동일하다는 조건이 공식적으로 포함되었기 때문이다. 본 연구에서 직접 실험을 한 결과, 어절의 수가 지나치게 많으면 탐지될 수 있는 기사의 수가 지나치게 줄어들고, 반대로 어절의 수가 지나치게 적으면 표절과 상관없는 기사들이 과다하게 탐지되었다.

3.3 OpenAPI 검색

OpenAPI는 사용자 및 개발자가 다양한 웹 서비스 및 응용을 개발할 수 있도록 기술과 서비스를 공유하는 프로그램이다. 웹 검색과 관련한 OpenAPI는 주로 구글 [4], 네이버, 다음과 같은 검색 포탈사이트에서 제공되고 있다. 본 연구에서는 AJAX 기반의 구글 API와는 달리 간단한 URL 조작에 기반한 네이버 및 다음 API를 활용하였다. 이러한 검색 API는 지식, 블로그, 웹문서, 뉴스 등등 여러 종류의 문서들에 대한 검색 기능을 제공하고 있는데, 본 논문에서는 네이버와 다음의 OpenAPI 중에서 뉴스 검색 API를 사용하였다.

그림 4와 그림 5는 각각 네이버와 다음에서 제공하는 OpenAPI의 검색 요구 URL을 보여준다.

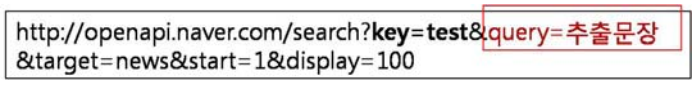

그림 4 네이버 OpenAPI 검색 요구 URL

그림 5 다음 OpenAPI 검색 요구 URL

검색은 Naver, Daum 중에 하나를 선택하여 할 수 있거나 또는 모두 활용하여 할 수 있다. 그림 4와 그림 5에서와 같이 각각의 추출된 문장은 query부분에 들어간다.

3.4 검색 결과 URL 추출

3.3에서 보여진 형태의 URL을 요청하면 응용에서 검색엔진에게 요청하면 그 결과는 그림 6과 같은 XML문서 형태로 전달된다. 그림 6은 요청된 결과인 XML문서의 예이다.

그림 6 요청 결과 XML 문서 사례

전달된 XML문서는 파싱을 하여 <originallink> .. </originallink> 또는 <LINK> .. </LINK> 부분 즉, 해당기사의 URL만을 추출한다. 하나의 원본 기사에서는 총 5개의 XML 문서에서 5개의 URL 집합을 구할 수 있다.

3.5 중복 URL 계산

이 단계에서는 3.4에서 구한 URL 집합에서 각 URL들이 몇 개의 집합에 포함되는지를 계산한다. 즉 다섯 개의 질의 문장 중에서 2개 이상의 문장을 포함하는 기사의 URL을 찾아내 사용자에게 표절 가능성이 높은 기사로 제시하게 된다. 그림 7은 Naver와 Daum 모두를 사용하여 탐지된 URL과 그 URL이 포함하고 있는 질의 문장의 수를 보여준다.

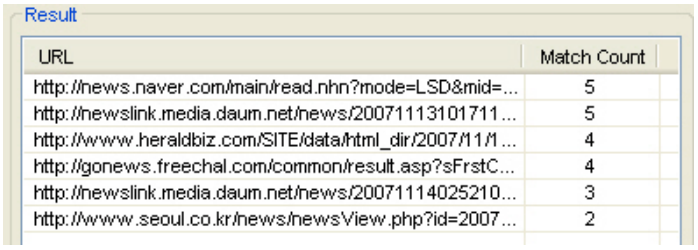

그림 7. 표절 기사 탐지 결과

여기서 첫 번째 열은 해당 URL을 그리고 두 번째 열은 그 URL이 포함하고 있는 추출문장의 수를 의미한다. 사용자는 이 결과를 토대로 개별 URL이 어느 정도 표절의 가능성을 가지고 있는가를 판단할 수 있으며 최종 판단은 해당 페이지를 직접 열어서 할 수 있다.

[4] http://code.google.com

4. 구현 및 실험

4.1 구현

본 시스템의 구현은 C#으로 작성했으며 사용 운영체제는 Windows XP SP2이다. 인터넷을 이용하기 때문에 인터넷이 항상 연결되어 있어야 한다.

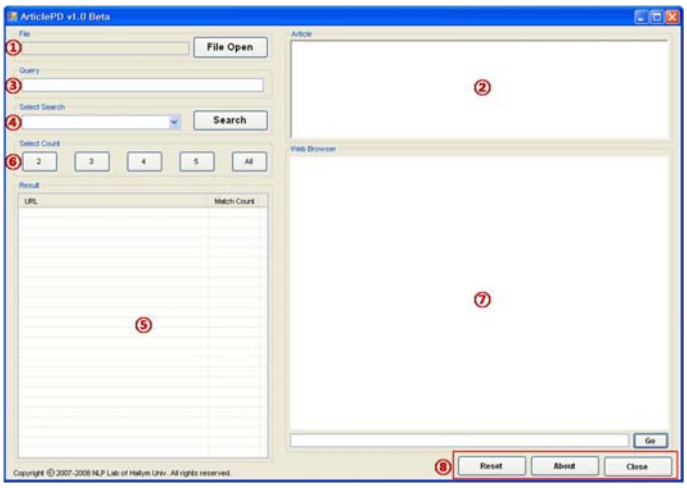

그림 8. 시스템 UI

그림 7은 본 시스템의 UI를 보여준다. ①은 원본 기사의 텍스트 파일을 입력하는 부분이다. 텍스트 파일의 종류는 확장자가 TXT인 파일만 가능하다. 파일을 열게 되면 ②에 원본기사의 내용을 출력한다. ③은 검색할 문장(Query) 하나를 입력하는 부분이다. 이것은 수동으로 한 문장을 검색할 때 사용한다. ④는 어떤 검색 OpenAPI를 사용할 것인지 선택하는 부분이다. 선택은 네이버, 다음 또는 모두 선택할 수 있다. 선택한 후 Search버튼을 누르면 ②의 원본 기사 내용에서 선택된 다섯 문장을 빨간색으로 나태 내며 검색하여 탐지된 결과를 ⑤의 리스트박스에 보여준다. 결과는 탐지된 URL과 그 URL의 중복 횟수를 보여준다. ⑥의 버튼은 결과에서 중복횟수 별로 보고자 할 때 사용한다. ⑦은 ⑤의 결과에서 URL을 더블 클릭 하면 그 해당 기사의 사이트를 보여주는 인터넷 창이다. 또한 7의 밑에 주소 창과 이동 버튼을 제공한다.

⑧은 모든 창을 초기화 하는 Reset 버튼, 프로그램을 닫는 Close 버튼, 프로그램 정보를 보여주는 About버튼 이다.

4.2 실험

본 시스템의 실험을 위한 원본 기사는 연합뉴스에서 발췌하여 사용하였다. 이 원본 기사는 다양한 분야에서 무작위로 선택한 것이고 개수는 총 30개 이다. 이 30개의 기사를 네이버 검색, 다음 검색으로 각각 나누어 실험하였다. 시스템을 통해 나온 결과인 표절 가능성이 있는 기사를 표절유형1, 2, 3의 기준으로 원본 기사와 직접 비교하여 표절기사 인지 판별하고 표절기사와 표절이 아닌 기사를 얼마나 탐지했는지를 비교해 웹 검색을 활용한 기사 탐지 시스템을 평가한다.

표 1. 네이버 검색 결과 기사 개수

	표절유형1	표절유형2	표절유형3	원본기사	표절이 아닌 기사	합계
2회		6	12		37	55
3회		3	10		12	25
4회	3	2	8	2	1	16
5회		2	4	24	1	31
합계	3	13	34	26	51	127

표 2. 중복 횟수 별 표절기사 비율(네이버)

	2회	3회	4회	5회	전체
표절기사 비율	32.7%	52.0%	92.8%	85.7%	49.5%

표 1과 표 2는 네이버 검색을 선택한 실험 결과 이다. 표절 기사는 2장에서 설명한 세 개의 기사 표절 유형과 검색이 중복되는 횟수 별로 나누었다. 표 1에서 원본 기사 항목은 원본 기사가 탐지된 것이거나 출처를 알리는 크래딧을 달아놓은 기사이다. 30개의 원본 기사를 통해 시스템에서 탐지한 기사는 총 127(표 2)개 이다. 이 중 원본기사를 제외한 101(표 2)개의 기사 중 표절 기사는 50(표 2)개로 전체의 49.5%(표 2)이다. 이는 탐지된 기사의 절반 정도가 표절 기사로 사람이 직접 비교 할 때 보다 훨씬 적은 시간으로 간단하게 표절 기사를 찾을 수 있다는 것을 보여준다. 표 2의 중복 횟수 별 표절기사 비율은 원본 기사를 제외한 비율이다. 2회 중복은 표절 기사의 비율이 32.7%(표 2)로 낮지만 중복 횟수가 증가할수록 표절 기사의 비율이 높아지는 것을 볼 수 있다. 이는 중복 횟수가 높을수록 탐지된 기사의 표절 가능성이 높다는 것을 알 수 있다.

표 3. 다음 검색 결과 기사 개수

	표절유형1	표절유형2	표절유형3	원본기사	표절이 아닌 기사	합계
2회		5	8		30	43
3회		5	11		8	24
4회		3	3	6	2	14
5회			1	24		25
합계	0	13	23	30	40	106

표 4. 중복 횟수 별 표절 기사 비율(다음)

	2회	3회	4회	5회	전체
표절기사 비율	27.9%	66.6%	75.0%		47.3%

표 3과 표 4는 다음 검색을 선택한 실험 결과 이다. 탐지된 총106(표 3)개의 기사 중 원본기사를 제외한 76개(표 3)의 기사 중 표절기사는 36개(표 3)로 전체의 47.3%(표 4)이다. 역시 네이버 검색 시와 비슷하게 절반 정도의 표절 기사를 탐지하는 것을 볼 수 있고 표 4에서와 같이 중복 횟수가 증가할수록 표절기사의 비율이 높아지는 것을 볼 수 있다. 5회 중복은 표절기사가 단 한번만 나왔기 때문에 비율은 넣지 않았다.

5. 결론 및 향후 연구방향

본 논문에서는 기사의 표절을 효율적으로 탐지해 내기 위한 웹 검색을 활용한 기사 표절 탐지 시스템을 제안하였다. 기사의 특성을 반영하기 힘든 기존연구에 비해 인터넷의 장점을 이용, OpenAPI를 활용해 기사를 찾아낸다. 본 시스템에서 탐지된 표절 가능성이 있는 기사의 약 48.4%가 표절기사였다. 이는 간단하게 한번의 탐지로 짧은 시간에 많은 표절 기사를 찾아줌으로써 시간, 인력, 비용을 크게 단축할 수 있는 것이다.

향후 연구는 표절 기사를 더 정확하게 탐지하기 위해서 원본 기사에서 문장을 추출하는 방법을 개선하고 Google의 OpenAPI도 적용하여 앞서 사용한 다른 OpenAPI와 비교해 본다. 또한 기사뿐만 아니라 블로그의 표절 탐지에도 적용해 본다. 마지막으로 현재 탐지 시스템의 결과에서 더 나아가 탐지된 기사의 표절 정도를 자동으로 판단할 수 있는 시스템 구현을 목표로 한다.

Acknowledgements

본 연구는 산업자원부의 지역혁신 인력양성사업(KOTEF)의 지원으로 이루어졌습니다.

참고 문헌

[1] 손정우, 박성배, 이상조, 박세영, “Parse Tree Kernel을 이용한 소스코드 표절 검출,” *한국컴퓨터종합학술대회 논문집*, Vol. 33, No. 1(B), 2006.

[2] Stefan Gruner, Stuart Naven, “Tool support for plagiarism detection in text documents,” *Proceedings of the 2005 ACM symposium on Applied computing*, DE, pages 776-781, 2005

[3] 류창건, 김형준, 박병준, 최혜정, 조환규, “한글 말뭉치를 이용한 한글 표절 탐색 모델 개발,” *한국정보과학회 학술발표 논문집*, 제34권 제2호(A), pages 58~59, 2007.

[4] 김지수, “OMUCS와 서열 정렬 기법을 이용한 영어 텍스트 표절 탐색 시스템의 설계 및 구현,” 중앙대학교 석사학위논문, 2005.

[5] 전명재, “대용량 한글 문서를 위한 표절 검색 시스템 개발,” 부산대학교 석사학위논문, 2005.